MA 3232 - Numerical Analysis
Objectives and Course Overview

I. Introduction

This course responds to the needs of the engineering and physical sciences curricula by providing an applications-oriented introduction to numerical methods/analysis. Rather than a pure discussion and analysis of methods, we shall often proceed from engineering and physical problems through a discussion of methods by which such problems may be solved numerically. This sequence is more \natural" and more like the one students actually follow when applying numerical methods within their areas of interest. For example, a simple resistance network with an applied voltage leads to a system of linear equations, and to a discussion of alternative methods for their computer solution and the costs and reliability of each.

Topics in nonlinear equations, eigenvalues and eignevectors, interpolation, numerical integration and differentiation, and numerical solution of ordinary and partial differential equations will be similarly treated. The discussion of approximate arithmetic and error propagation will also arise in a natural way.

II. Objectives

Upon successful completion of this course, one should be able to:

1. Describe difficulties that can arise because computers usually use non-decimal arithmetic with finite length representation.
2. List sources of error in computation, particularly to be able to identify when catastrophic cancellation may occur in a given computation.
3. Apply the methods of bisection, false position, secant, and Newton to solve a given nonlinear equation.
4. State conditions under which the method of iteration ($x_{n+1} = g(x_n)$) converges and show it converges in a given case.
5. Apply the Newton-Raphson method to solve systems of nonlinear equations.
6. Construct Lagrange and Newton-Gregory forward difference interpolation polynomials for a given set of data.
7. State and use the formulas for bounding the error in polynomial interpolation based on derivatives, and for estimating error based on differences.
8. Describe the basic ideas and principles behind cubic spline interpolation, and its limitations.
9. Construct and apply formulas for approximating specified derivatives of functions by differentiation of the appropriate Newton-Gregory interpolation polynomial. Estimate the error in calculated derivative approximations.
10. Construct quadrature formulas with evenly-spaced nodes by integration of the interpolation polynomial.
11. Apply trapezoidal and Simpson's Rule quadratures in composite form to find the approximate value of an integral. Estimate the errors in computed quadratures and use these estimates to improve the original values.
12. State the idea behind the derivation of Gauss quadrature rules, and why they often work very well. Apply given low order Gauss quadrature.

13. Describe the basic concepts behind Runge-Kutta (R-K) methods for the initial value problem (IVP) and apply specific R-K methods in given problems.

14. State how multistep methods are derived and apply multistep predictor-corrector methods in given problems.

15. State the advantages/disadvantages of single step and multistep methods for the IVP and when use of each is recommended.

16. Explain the possible instability of the multistep methods.

17. State the definition of the order of an interpolation, differentiation, or integration method, and describe the relationships between the order of a given method and error estimation.

18. Describe the basic concepts behind and differences between shooting, finite difference, Galerkin, finite-element, and Rayleigh-Ritz methods for ordinary differential equation boundary value problems, and apply each in appropriate problems.

19. Describe what is meant by the error, the residual and the condition number in the context of numerical solution of systems of linear equations, i.e. $Ax = b$ and describe the relationship between them.

20. Describe and use the Jacobi, Gauss-Seidel and Successive Overrelaxation (SOR) iterative methods for solving $Ax = b$; explain when such methods are used; give sufficient conditions for convergence; and tell why SOR is usually superior to Gauss-Seidel, which in turn is usually superior to Jacobi.

21. Describe the basic concepts behind the normal equations for least squares, and the limitations of the method. Apply the normal equations to solve a given overdetermined system of linear equations.

22. Compute low degree polynomial least squares curve fits for given data.

23. Use power method iteration either to find the largest eigenvalue of a matrix or to refine estimates of other eigenvalues. Describe when any why the method should converge.

24. State the definition of an orthogonal matrix, and why such matrices are important in numerical analysis.

25. Formulate the explicit finite difference equations for a given parabolic, elliptic or hyperbolic partial differential equation.

26. Describe the difference between implicit versus an explicit formulation of a partial differential equation, and under what general conditions implicit formulations may be superior. Specifically describe the Crank-Nicholson formulation for the one-dimensional heat equation, and its advantages and disadvantages.

27. Describe what the terms convergence, stability and consistency mean in the context of numerical solution of partial differential equations.

28. Given library routines for the solution of the above problems as well as other utility routines, be able to indicate how to use them in the solution of a larger problem involving the solution of several 'sub-problems'.

    For example:

    Assume that routines for the solution of (1) a nonlinear equation and (2) an ordinary differential equation on the interval $[a; b]$, as well as hardware and software for producing (3) appropriate graphical output (e.g. curve plots). (Specific information about input, output, and other requirements for these routines will be given.) The student may then be asked to outline (in English, MATLAB, flow charts, mathematical symbols, or a combination thereof) a program for solution of the following problem:

With $y(x)$ defined by

$$y' = f(x, y)$$
$$y(a) = y_0,$$

find the smallest $x^* > a$ such that $y(x^*) = 0$, and plot y versus x on $[a, x^*]$.